REMARKS

Specification

Applicant has amended the Specification to include patent application serial numbers and a patent number.

Claims

Claims 2 to 5 were pending when last examined. With this Response, Applicant amends claims 2 to 5.

The Examiner rejected claims 2 to 5 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,374,331 ("Janakiraman et al.") in view of U.S. Patent No. 6,513,142 ("Noya"). Applicant has amended claim 2 as follows:

> 2.   A method for communicating data in a data storage system, the data storage system comprising a plurality of interconnected nodes, each node having a respective cache memory comprising a plurality of cache lines, each cache line having the same predetermined size, the method comprising:
>
> > providing new data for writing into a portion of a particular cache line in a memory region of a cache memory located at a local node, wherein data written to the memory region are mirrored to at least another memory region in at least another cache memory located at a remote node;
> >
> > reading out existing data from the particular cache line in the memory region of the cache memory located at the local node;
> >
> > merging the new data with the existing data;
> >
> > writing the merged data into the particular cache line in the memory region of the cache memory at the local node; and
> >
> > transferring the merged data over a communication link to the remote node for writing into said another memory region in said another cache memory located at the remote node.

Amended claim 2.

Addressing original claim 2, the Examiner stated:

> 4.   As to claim 2, Janakiraman discloses the invention substantially as claimed, including a method for communicating data in an interconnect system (col. 2, lines 51-55) comprising a plurality of nodes (200-500, fig. 1), each node

-5-

having a respective memory (210-410, fig. 1) comprising a plurality of lines (col. 1, lines 41-44 and 59-61; col. 3, lines 46-47), each line of the same predetermined size (i.e., a portion of memory tens of bytes in size; col. 1, lines 41-44), the method comprising:

> providing new data for writing into a portion of a particular line of memory located at a local node (col. 5, lines 15-18; col. 6, lines 54-60); and

> transferring the data over a communication link (col. 2, lines 53-55) to a remote node for writing into memory located at the remote node (col. 2, lines 33-38; col. 3, lines 14-16; col. 4, lines 36-50).

November 9, 2004 Office Action, ¶ 4. Applicant respectfully traverses.

Janakiraman et al. does not disclose "transferring the merged data over a communication link to a remote node for writing into memory located at the remote node" as recited in the original claim 2, which remains essentially the same in amended claim 2. For this limitation, the Examiner cited Janakiraman et al. at col. 2, lines 33 to 38; col. 3, lines 14 to 16; and col. 4, lines 36 to 50. Janakiraman et al. at col. 2, lines 33 to 38 states,

> When a node requires access to data that is in a cache or memory in another node's processor, a message must first traverse the network from the requesting node to the node maintaining the directory. Then, the node maintaining the directory must send another message to the node with the data.

Janakiraman et al., col. 2, lines 33 to 38. The quoted lines provide that when a node (e.g., a local node) wants to read data in cache or memory located elsewhere in the system, the local node sends a request to another node (e.g., a remote node) that maintains the coherence of that cache or memory. In response, the remote node sends a message with the requested data back to the local node. On the contrary, claim 2 recites that the local node sends merged data to the remote note to write to the memory of the remote node.

Janakiraman et al. at col. 3, lines 14 to 16 states, "Fig. 4 is the multi-processor computer system of the present invention while accessing data maintained in a remote memory unit." Fig. 4 illustrates a local node 2000 accessing a remote memory at a remote memory 3010.

> Referring now to FIG. 4, therein is shown the ccNUMA system 1000 when the access to a memory line that is maintained in a remote memory, such as the memory unit 3010. The processor 2040 (or its associated network interface) sends a request for access to the communication switch and coherence controller maintaining the directory entry for that memory line, such as the communication switch and coherence controller 6000. On receiving the request, along the route indicated by the arrow 2060, the communication switch and coherence controller

-6-

Serial No. 09/751,994

6000 looks up the directory entry in the coherence directory 6030 associated with the requested memory line. If the directory entry indicates that the value maintained in memory is valid, a message is sent to the node, such as the node 3000, that maintains the memory unit 3010 containing the memory line. The node 3000 reads the requested memory line from its memory unit 3010 and sends it in a data message along the route indicated by the arrow 2080 to the requesting node 2000. This data message is routed through the communication switch and coherence controller 6000 which maintains the coherence directory 6030, allowing it to identify that the access has been completed.

Janakiraman et al., col. 5, line 64 to col. 6, line 18. Similar to above, the quoted lines and related description provide that when a local node wants to read data in a remote memory, the local node sends a request to a communication switch/coherence controller that maintains the coherence for the remote memory. In response, the communication switch/coherence controller sends a message to the remote node that maintains the memory. The remote node then sends the requested data through the communication switch/coherence controller back to the local node. On the contrary, claim 2 recites that the local node sends merged data to the remote note to write to the memory of the remote node.

Janakiraman et al. at col. 4, lines 36 to 50 states,

A coherence architecture where coherence directories and control are located at the site of memory unit may also result in longer access to remote data when nodes are situated at the end points of the network. Commercial systems have typically been built in hierarchical topologies where the nodes are at the leaves, or end points, of the network. In these systems, memory and coherence directories are located at the end points of the network. When a node requires access to data that is in a cache in another node's processor, the message must first traverse the network from the requesting node to the node maintaining the directory. Then, the node maintaining the directory must send another message to the node with the data. Finally, the data must flow from the node with the data to the node requesting the data.

Janakiraman et al., col. 4, lines 36 to 50 (emphasis added). Again, the quoted lines provide that when a local node wants to read data in a remote memory, the local node must send a request to a remote node that maintains the coherence of that memory and in response the remote node sends a message with the requested data back to the local node. On the contrary, claim 2 recites that the local node sends merged data to the remote note to write to the memory of the remote node.

Furthermore, claim 2 recites that a portion of the cache line at the local memory is provided with new data but the entire cache line is rewritten with the result of merging the new data and the existing data in the entire cache line. Janakiraman et al. discloses accessing a memory line but not writing to a portion of the memory line. "The communication switch services read and write requests

-7-   Serial No. 09/751,994
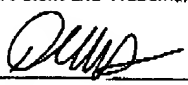
to these memory lines (as does the home node in the traditional architecture shown in FIG. 1 (PRIOR ART)). Janakiraman et al., col. 5, lines 15 to 18. Noya discloses writing only the new data, and not the result of margining the new data and the existing data, to a block in a disk array implementing RAID 5. "Referring also to FIG. 5, the new data is then written into the disk (via the cache) as updated block 2' (step D)." Noya, col. 5, lines 38 to 40.

In addition, claim 2 now recites "a data storage system ... wherein data written to the memory region [at a cache memory at a local node] are mirrored to at least another memory region in at least another cache memory located at a remote node." Neither Janakiraman et al. nor Noya discloses a data storage system with such a mirroring scheme.

Accordingly, claim 2 is patentable over the combination of Janakiraman et al. and Noya.

Claims 3 to 5 depend from claim 2 and are patentable over the cited references for at least the same reasons as claim 2.

In summary, claims 2 to 5 were pending in the above-identified application when last examined. For the above reasons, Applicant respectfully requests the Examiner with withdraw the claim rejections and allow claims 2 to 5. Should the Examiner have any questions, please call the undersigned at (408) 382-0480x206.

Certification of Facsimile Transmission
I hereby certify that this paper is being facsimile transmitted to the U.S. Patent and Trademark Office on the date shown below.
Signature _____ Date 2/9/05

Respectfully submitted,

David C. Hsia
Attorney for Applicant(s)
Reg. No. 46,235

-8-

Serial No. 09/751,994

PAGE 9/10 * RCVD AT 2/9/2005 7:47:18 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-1/1 * DNIS:8729306 * CSID:4083820481 * DURATION (mm-ss):03-24